11-18

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)

2. REPORT DATE

3. REPORT TYPE AND DATES COVERED

4. TITLE AND SUBTITLE

Object Representaiton for Design-Unifying Cubes and Spheres

5. FUNDING NUMBERS

DAAL03-87-K-0005

6. AUTHOR(S)

Williams, John R. and Pentland, Alex P.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Massachusetts Institute of Technology
Center for Construction Research and Edcuation
Technology and Development Program

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U. S. Army Research Office
P. O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

ARO 24620.22-EG-UIR

11. SUPPLEMENTARY NOTES

The view, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

The way in which we represent physical objects in the computer greatly affects our ability to manipulate and reason about them. We are concerned with developing computer tools for conceptual deisgn, and seek representations which are appropriate for both CAD and finite element calculations. Here, we hgihlight the use of superquadric functions for representing object surfaces and volumes. Using this representaiton bodies can be created as generic 'blobs' and then molded, rather like clay, to the reuqired shape usign easily controlled parameters. These functions have attractive properties, including being able to preresent angular bodies, such as cubes, with a smooth continuously differentiable surface. It is demonstrated that superquadrics can also be used to generate potential surfaces for plasticity. These do not have the traditional 'corner problems' for surfaces, such as Tresca. Super-quadrics can also be ocmbined with the finite element representations to provide an efficient analysis scheme for calculating the dynamics of multi-body systems.

SUBJECT TERMS

Virtual Construction; conceptual design; CAD; Finite element analysis; Multi-body dynamics.

15. NUMBER OF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

COMPLAS '89
2nd International Conference on
Computational Plasticity
Barcelona, Spain
18-22 September, 1989

OBJECT REPRESENTATION FOR DESIGN – UNIFYING CUBES AND SPHERES

John R. Williams and Alex P. Pentland*

INTELLIGENT ENGINEERING SYSTEMS LABORATORY
CIVIL ENGINEERING DEPARTMENT, 1–272
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MA 02139

ABSTRACT

The way in which we represent physical objects in the computer greatly affects our ability to manipulate and reason about them. We are concerned with developing computer tools for conceptual design, and seek representations which are appropriate for both CAD and finite element calculations. Here, we highlight the use of superquadric functions for representing object surfaces and volumes. Using this representation bodies can be created as generic 'blobs' and then molded, rather like clay, to the required shape using easily controlled parameters. These functions have attractive properties, including being able to represent angular bodies, such as cubes, with a smooth continuously differentiable surface. It is demonstrated that superquadrics can also be used to generate potential surfaces for plasticity. These do not have the traditional 'corner problems' for surfaces, such as Tresca. Superquadrics can also be combined with finite element representations to provide an efficient analysis scheme for calculating the dynamics of multi–body systems.

1 DESIGN TOOLS

There is a growing belief in the U.S. that the present design process must be significantly improved. In particular, we must reason about issues, such as manufacturing and maintenance, during the early design

---

* This author is with the Vision Sciences Group, Media Laboratory, E15–410, MIT

† This work was made possible by the U.S. Army Research Office URI Program for Advanced Construction Technology Grant #DAAL–03–87–0005 and by a grant from Nippon Telephone and Telegraph.

stage; an approach which has been called 'Concurrent Design' [1].

'Concurrent Design is a systematic approach to the integrated, concurrent design of products and their related processes, including manufacture and support. This approach is intended to cause developers, from the outset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, and user requirements'.

Here we discuss the use of interactive simulation of multi–body dynamics for 'Virtual Manufacturing' – the simulation of the manufacturing process in the computer during the early stages of design. In contrast to present CAD and finite element analysis tools, which deal with design detailing and with the optimization of the design after the major design decisions have been made, conceptual design deals with an ill defined product, and must provide guidance for selecting between radically different concepts. The system we envisage provides visual feedback to the designer on product manufacturing processes and on performance during testing.

If computer tools are to replace pencil, paper and calculator, we need to be able to interact with them with similar ease. The human–machine interface problem in design is the topic of our ongoing research and is discussed in other papers [2,3]. Here we concentrate on the problem of analyzing and reasoning about spatial relationships of multi–body systems, including contact and interference. In common with previous attempts at achieving this goal, we have been confronted with the problem that, the computational expense of calculating dynamic interactions, prevents interactive simulations for all but relatively small problems. We believe that fundamental changes in the way we represent objects may lead to improvements in this area.

1.1 Simulation

Since virtual manufacturing requires that we reason about the effectiveness of the design from the very start of the project, we must have numerical analysis tools integrated into our design system. The computer simulation of the multi–body dynamics of deformable objects has long been a major goal of researchers in computational mechanics [4,5,6] and computer graphics [7,8,9]. In computational mechanics the emphasis has been on the detailed analysis of relatively few (typically two) impacting bodies, where the need for accuracy has been of paramount importance. The emphasis in computer graphics, in projects such as Sketchpad [7] and Thinglab [8], has been on producing realistic animation of colliding objects.

In Discrete Element [10,11] problems with large numbers of bodies, as much as 80% of the computational time is spent in detecting and tracking the contacts between bodies. The problem is especially severe because the algorithm is of order O(nm) operations, where n is the number of polygons and m is the number of points to be checked for

interference. Some finite element implementations [12] avoid automated checking by having the user input contacting surfaces, but this is only possible when these surfaces can be predicted a priori. Here we consider the general problem of reasoning about spatial relationships, including contact detection, when they must be resolved by the computer program.surfaces. The method we present uses a hybrid approach in which the geometry of the object is represented by superquadric functions while for the stress calculations the displacements are interpolated using standard parabolic shape functions.

The use of different representations for surface and internal geometry focuses our attention on a field of increasing importance – object representation. In the following sections we discuss how we have been able to take advantage of volumetric object representation based on superquadric functions by combining it with a reduced basis modal approach for calculating object deformations. Examples of the use of this approach for design are presented in the context of a prototype program called Thingworld [13].

2 SUPERQUADRICS AND HYPERQUADRICS

2.1 Superquadrics

Superquadrics and hyperquadrics [14,15,16] are examples of representations where a single analytic expression defines the complete bounding surface of an object. The general equation of a three dimensional superquadric is given by:

$$\left|\frac{x}{a}\right|^{\epsilon_1} + \left|\frac{y}{b}\right|^{\epsilon_2} + \left|\frac{z}{c}\right|^{\epsilon_3} = 1 \qquad \text{Eqn. 1}$$

where a,b and c, determine the lengths of the prinicple axes and ϵ_1, ϵ_2 and ϵ_3 are powers, such that $0 < \epsilon < \infty$.

The properties of the superellipse or superquadric equation were first investigated by the Danish designer Peit Hein [17]. When $\epsilon = 2$ we recover the equation of an ellipsoid. By varying a,b and c we can stretch the body continuously in any direction. By varying ϵ from infinity to zero we obtain the curves shown in Figure 1.

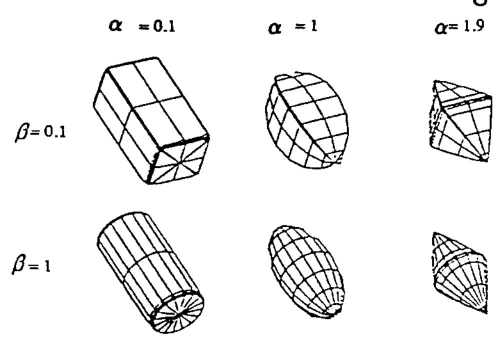

Figure 1 Superquadric Functions For Various Parameter Values.

A particularly useful facet of volumetric expressions is the so called 'inside–outside' property. To determine if the point (x ,y) is inside or outside the body we substitute the point into the function below:

$$F(x,y,z) = \left|\frac{x}{a}\right|^{\epsilon_1} + \left|\frac{y}{b}\right|^{\epsilon_2} + \left|\frac{z}{c}\right|^{\epsilon_3} - 1 \qquad \text{Eqn 2}$$

Now if F, is greater than zero the point is outside the body, if F is equal to zero the point is on the surface, and if F is less than zero the point is inside the body. Furthermore, we note that for $\epsilon = 2$, $|F|$ is the square of the distance of the point from the surface. In general Equation 2 provides a non–Riemann measure of the distance of the point (x,y,z) from the surface of the superquadratic, $F(x,y,z) = 0$.
For $\alpha = \beta = 2/n$, F(x,y,z) gives the L_n distance metric. Thus, the superquadric can be viewed as specifying a family of potential surfaces throughout space, one of which represents the object surface.

The modulated superquadric equation can be written as,

$$\left[\left|\frac{x}{a}\right|^{2/\beta} + \left|\frac{y}{b}\right|^{2/\beta} \right]^{\beta/\alpha} + \left|\frac{z}{c}\right|^{2/\alpha} = 1 \qquad \text{Eqn. 3}$$

The superquadric family can also be represented parametrically by latitude and longitude parameters as:

$$\vec{R}(\eta,\omega) = \begin{bmatrix} a\ \mathrm{sign}(\cos\varphi\cos\theta)\ |\cos^{\beta}(\varphi)\ \cos^{\alpha}(\theta)| \\ b\ \mathrm{sign}(\cos\varphi\sin\theta)\ |\cos^{\beta}(\varphi)\ \sin^{\alpha}(\theta)| \\ c\ \mathrm{sign}(\sin\varphi)\ |\sin^{\beta}(\varphi)| \end{bmatrix} \qquad \text{Eqn 4}$$

Where $\vec{R}$ is a three dimensional vector containing the x, y, z, components of the surface point. The normal to the surface at point (φ,θ) is given by:

$$\vec{N}(\eta,\omega) = \begin{bmatrix} 1/a\ \mathrm{sign}(\cos\varphi\cos\theta)|\cos^{2-\beta}(\varphi)\ \cos^{2-\alpha}(\theta)| \\ 1/b\ \mathrm{sign}(\cos\varphi\sin\theta)|\cos^{2-\beta}(\varphi)\ \sin^{2-\alpha}(\theta)| \\ 1/c\ \mathrm{sign}(\sin\varphi)\ |\sin^{2-\beta}(\varphi)| \end{bmatrix}$$

The superquadric has been exploited in shape recovery from digital images by Solina [18], who uses the following modified 'inside– outside' function to implement a least squares approach.

$$F(x,y,z) = \left[\left[\left[\frac{x}{a}\right]^{2/\beta} + \left[\frac{y}{b}\right]^{2/\beta} \right]^{\beta/\alpha} + \left[\frac{z}{c}\right]^{2/\alpha} \right]^{\alpha}$$

This modified function gives a more consistent quadratic distance measure to points deviating from the superquadric surface.

The inside–outside function defines the superquadric surface in a body centered co–ordinate system. To locate the body at an arbitrary position in global co–ordinates, points are transformed using a homogeneous co–ordinate transformation T, where T is of the form:

$$[T] = \begin{bmatrix} n_x & m_x & l_x & d_x \\ n_y & m_y & l_y & d_y \\ n_z & m_z & l_z & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and n,m,l are the direction cosines of the local body axes with respect to the global axes, and **d**, is the translation vector to the local body axes origin.

Thus,

$$[T]\{\mathbf{x}_{local}\} = \{\mathbf{x}_{world}\}$$

where $\{\mathbf{x}\}^{Tr} = \{x, y, z, 1\}$.

Deformations, not directly controlled by superquadric parameters, can be imposed by using Jacobians to deform physical space, eg

$$\{\mathbf{x}^*\} = \begin{bmatrix} 1 & 0 & k & 0 \\ 0 & 1 & k & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\{\mathbf{x}\}$$

defines a form tapered along the z axis. The corresponding transformation for the surface normal (which transforms as a covariant tensor, as opposed to a contravariant tensor) is given by:

$$\vec{N}^* = 1/(\det J)*(J^{-1})^T \vec{N}$$

where J is the Jacobian of the deformation mapping.

2.2 Hyperquadrics

By analogy with Equation 1, we can expand the family of superquadrics to higher numbers of dimensions, so called hyperquadrics. We note the equation may be written for four dimensions as:

$$F(x,y,z,w) = \left|\frac{x}{a}\right|^{\epsilon_1} + \left|\frac{y}{b}\right|^{\epsilon_2} + \left|\frac{z}{c}\right|^{\epsilon_3} + \left|\frac{w}{d}\right|^{\epsilon_4} - 1 \qquad \text{Eqn 5}$$

and for n dimensions as:

$$\sum_i^n |B_i(\mathbf{x})|^{\epsilon_i} = 1 \; ; \qquad \text{Eqn 6}$$

where $B_i(x) = \sum_{j=1}^n a_{ij} x_j + e_i$, where $x_1 = x$, $x_2 = y$, etc. and e is included to be completely general.

Hanson [16] has shown that this proceedure allows us to design hypercubes, and to intersect them with lower dimensional hyperplanes to generate complicated three dimensional objects. Here we shall only concern ourselves with four dimensional hypercubes cut by three dimensional hyperplanes, to give us three dimensional volumes. Because visualizing four dimensions is difficult we proceed by first examining the problem in one dimension less, ie. three dimensional superquadrics cut by two dimensional planes.

Consider the intersection of a plane with a superquadric cube, as illustrated in Figure 2. The equation of the three dimensional unit cube is given by:

$$|X|^{\epsilon} + |Y|^{\epsilon} + |Z|^{\epsilon} = 1 \quad \text{Eqn. 7}$$

where capitalized letters indicate global coordinates.

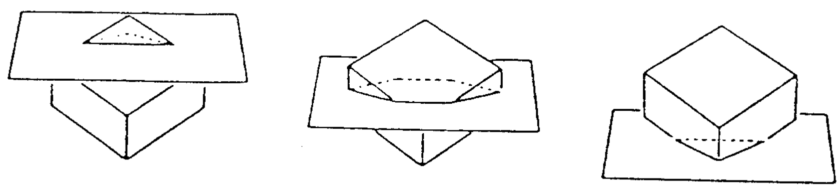

Figure 2 Intersection of Plane Through Cube

Let the cube be cut by a plane whose surface normal is **n**, so that its equation is:

$$\mathbf{n}^T . \mathbf{X} = d;$$

where $\mathbf{n}^T = \{n_1, n_2, n_3\}$. To find the equation of the intersecting surface, we transform global coordinates **X**, to a new local basis **x**, such that the orthonormal vectors of **x**, call them **a** and **b**, lie in the intersecting plane, ie. **n.a** = **n.b** = 0. The unit vectors **a** and **b**, can easily be found as follows:

Take any vector **c**, not parallel to **n**, and form **a** = **n** x **c**. Now take **b** = **a** x **n** and normalize both **a** and **b**. In terms of the new axes the old coordinates are given by:

$$\begin{bmatrix} x \\ y \\ d \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ n_1 & n_2 & n_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \text{ or } \{x\} = [\,S\,]\,\{X\}\,;$$

Since **a.a** = **b.b** = **n.n** = 1 and **a.b** = **a.n** = **b.n** = 0 and $[S]^{-1} = [S]^T$ it follows that:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & n_1 \\ a_2 & b_2 & n_2 \\ a_3 & b_3 & n_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ d \end{bmatrix} \qquad \text{Eqn 8}$$

Substituting this into Equation 4 gives the hyperquadric equation in terms of the new orthonormal coordinates.

$$\sum_{i}^{3} |B_i(\mathbf{x})|^{\epsilon_i} = \sum_{i=1}^{3} |a_i\, x + b_i\, y + n_i\, d|^{\epsilon_i} = 1 \quad \text{Eqn. 9;}$$

The parameter d specifies the distance from the origin to the plane of intersection. By varying d, we can cut the cube to form the shapes shown in Figure 3.

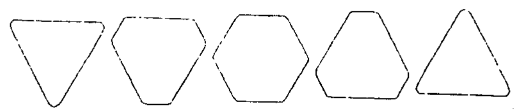

Figure 3 Curves Formed By Intersecting Cube With a Plane

If we parameterize Equation 9 in terms of θ and φ, so that:

$$\begin{aligned} x(\theta) &= r(\theta)\cos(\theta)\cos(\varphi), \\ y(\theta) &= r(\theta)\sin(\theta)\cos(\varphi), \\ d &= \sin(\varphi); \end{aligned}$$

we can solve numerically for $r(\theta)$, using a search technique, such as Newton–Raphson. We note that the asymptotic bounds on the allowed shapes are given by letting $\epsilon \to \infty$. Any value of B_i less than 1, tends to zero, because it is raised to a high power, leaving only points on the lines defined by each separate term B_i, given below:

$$a_i\, x + b_i\, y + n_i\, d = \pm 1\,;$$

By choosing bounding surfaces directly and creatoing an equation of the form of Equation 5, we can design surfaces which tend to various polygonal shapes.

2.3 Higher Dimensions

The equation of the unit hypercube in four dimensions is given by:

$$\sum_{i=1}^{4} |X_i|^{\epsilon_i} = 1\,;$$

The intersection of a hyperplane with normal $\mathbf{n}$, with the hypercube is given by

$$\sum_{i=1}^{4} |a_i\, x + b_i\, y + c_i\, z + n_i\, d|^{\epsilon_i} = 1\,;$$

where $\mathbf{a}.\mathbf{b} = \mathbf{a}.\mathbf{c} = \mathbf{a}.\mathbf{d} = \mathbf{b}.\mathbf{c} = \mathbf{b}.\mathbf{d} = \mathbf{c}.\mathbf{d} = 0$. Calculating a 4–vector orthogonal to three other 4–vectors, say $\mathbf{a}$, $\mathbf{b}$, $\mathbf{n}$ is accomplished by choosing the coefficients A, B, C, D as the co–factors of $\mathbf{i}$, $\mathbf{j}$, $\mathbf{k}$, and $\mathbf{l}$ in the determinant below;

$$\begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} & \mathbf{l} \\ a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ n_1 & n_2 & n_3 & n_4 \end{vmatrix} = A\mathbf{i} + B\mathbf{j} + C\mathbf{k} + D\mathbf{l}$$

where $\mathbf{l}$ is the unit vector in the 4th dimesnion.

In analogy with Equation 8 this gives the required transformation from global to local coordinates oriented normal to, and in, the hyperplane:

$$\begin{bmatrix} x \\ y \\ z \\ d \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ n_1 & n_2 & n_3 & n_4 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ D \end{bmatrix}$$

The corresponding parametric equations are:

$$\begin{aligned} x(\theta,\psi) &= r(\theta,\psi)\cos(\theta)\cos(\psi)\cos(\varphi), \\ y(\theta,\psi) &= r(\theta,\psi)\sin(\theta)\cos(\psi)\cos(\varphi), \\ z(\theta,\psi) &= r(\theta,\psi)\sin(\psi)\cos(\varphi) \quad , \\ d &= \sin(\varphi) \quad ; \end{aligned}$$

Figure 4 shows a number of three dimensional shapes formed by intersecting a hyperquadric with a hyperplane.

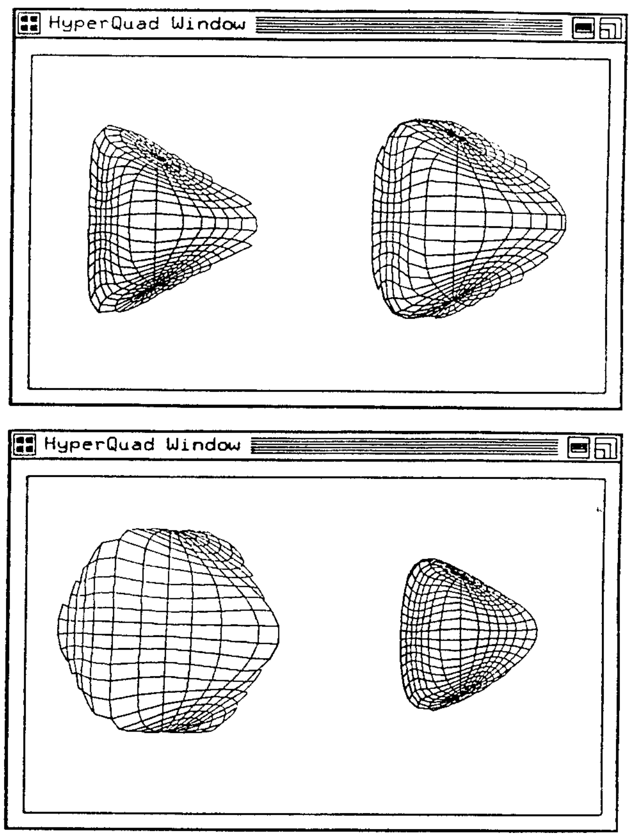

Figure 4 Hyperquadric Surfaces Formed By Varying Distance From Origin of Hyperplane

For plasticity calculations the normal to the surface is required. This is just the gradient of the equation at the given point, given by:

$$n_j = \frac{\partial B}{\partial x_j} = \sum_i \epsilon_i |B_i|^{\epsilon_i - 1} \operatorname{sign}(B_i)\, a_{ij}$$

where $B_i(x) = \sum_{j=1}^{n} a_{ij} x_j + d_i$

2.4 Application To Plastic Potentials

Many of the surfaces traditionally used in plasticity, such as Tresca, have non unique normals at their corners. We show here how to design surfaces to match these plastic potentials, which have unique normals at all points. Furthermore these surfaces are easily expanded to follow hardening and other material behavior.

Consider the boundary of the 'square' superquadric

$$|x|^{\epsilon} + |y|^{\epsilon} = 1$$

where ϵ is large. Let us add another term

$$|x+y|^{\epsilon} + |x|^{\epsilon} + |y|^{\epsilon} = 1$$

The figure is now bounded by the following lines:

$$x+y = \pm 1, \quad x = \pm 1, \quad y = \pm 1;$$

We note that whenever one of the terms has a value < 1 then its contribution becomes negligible, because it is raised to a large power. Thus by choosing the terms corresponding to $B_i(x)$, we can easily control the shape of the bounding surface. To generate an hexagonal boundary for, say Tresca, we choose

$$+0.707\, x - 0.408\, y + 0.577 = \pm 1,$$
$$-0.707\, x - 0.408\, y + 0.577 = \pm 1,$$
$$0.816y + 0.577 = \pm 1;$$

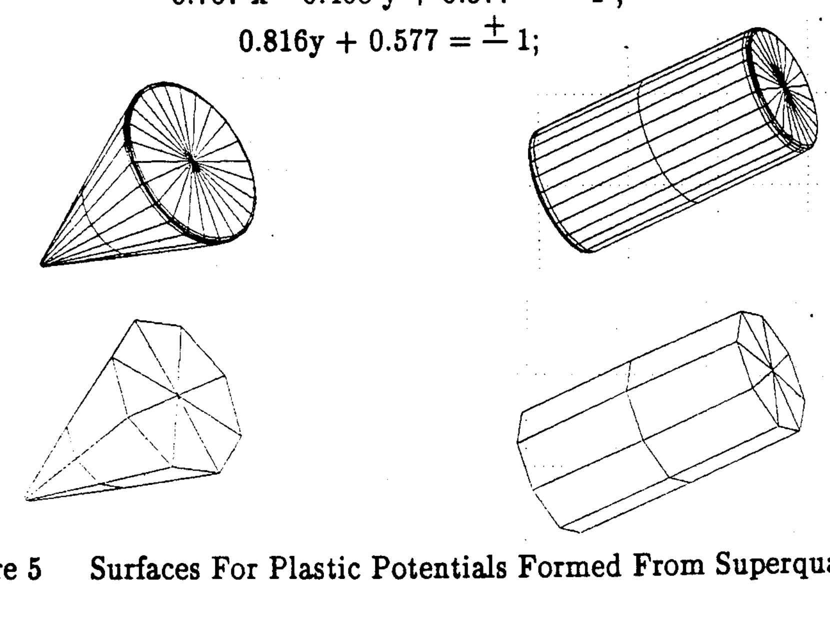

Figure 5 Surfaces For Plastic Potentials Formed From Superquads

If we wish to extrude this surface in three dimension we can add another term in the z direction. Figure 5 shows a number of surfaces generated using both superquadric and hyperquadric functions. Note that all the surfaces are actually smooth functions, with unique, well defined, normals everywhere, even though they are drawn as having sharp angles.

2.5 Contact Detection

The role of contact detection in multi–body dynamics is to determine the area of contact and the associated interaction forces between the bodies. There are two main methods of enforcing contact conditions, namely via a Lagrangian multiplier [18] or via a Penalty Function [19,20] formulation. Both methods essentially calculate a contact force (stress) which varies at each time step and depends on the relative geometries of the bodies. We show here how the 'inside–outside' functions enables efficient contact algorithms.

For rendering, and hidden line and surface calculations it is necessary to generate a set of points covering the surface of the superquadric. This is most easily achieved by using the parametric form of Equation 4 so that points are generated along lines of constant latitude and longitude. To check if a given object is in contact with any other, we first perform a bounding box check, which usually eliminates the majority of candidates for contact. We now loop over the points on our given object's surface and conduct an 'inside–outside' check with the superquadric function of the remaining objects. When we find a point inside another object, we intelligently search surrounding points until one outside is found. If necessary the intersection point, which lies between the inside and outside point, can be found by a search scheme, such as a binary search. Depending on the amount of effort we wish to expend we can now employ various methods of determining the area of contact. We are presently experimenting with both point contact and a method based on an average radius of curvature for the penetrating cap. The latter method is quite accurate when the local curvature is approximately constant.

3 MODAL DYNAMICS

The element dynamics in Thingworld are based on the eigenmodes of each body, as calculated using a parablolic interpolation. The parabolic element is determined by 20 points taken on the surface of the superquadric. These points link together the two representation, keeping them 'in–step'.

Two examples of analyses using the Thingworld system based on superquadrics are shown in Figures 6 and 7. The first analysis shows the dynamic impact of a ball on a two by four piece of wood. Constant, linear and quadratic strain modes were analyzed, allowing rapid calculation of the response. On a 1 Mip Symbolics machine each intersection calcultion took only 0.05 secs allowing the whole analysis to be completed (excluding rendering) in approximately real time.

Figure 7 shows a design analysis of a chair. We start with the seat which we mold to shape. Taking a similar piece we form the back. Adding tubes for legs and arms we assemble our chair in Thingworld in approximately 5 mins. Defining constraints to fix arms and legs in place takes another minute and then defining boundary conditions and loads another minute. Here we choose to have a person (from our library of parts) lie on the chair to test its strength and stability. This model took only 180 parameters to define, illustrating the compact representations available using superquadrics.

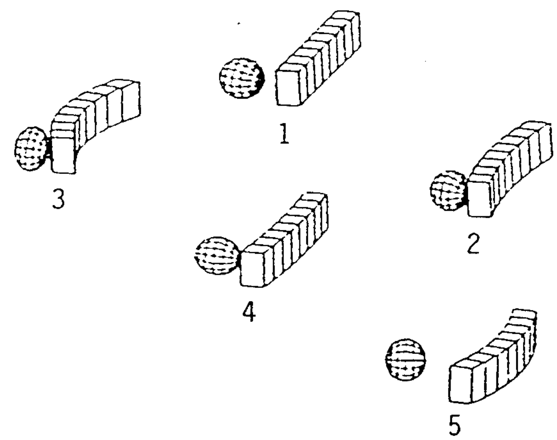

Figure 6 Dynamic Modal Analysis of a Deformable Ball Hitting a Two By Four

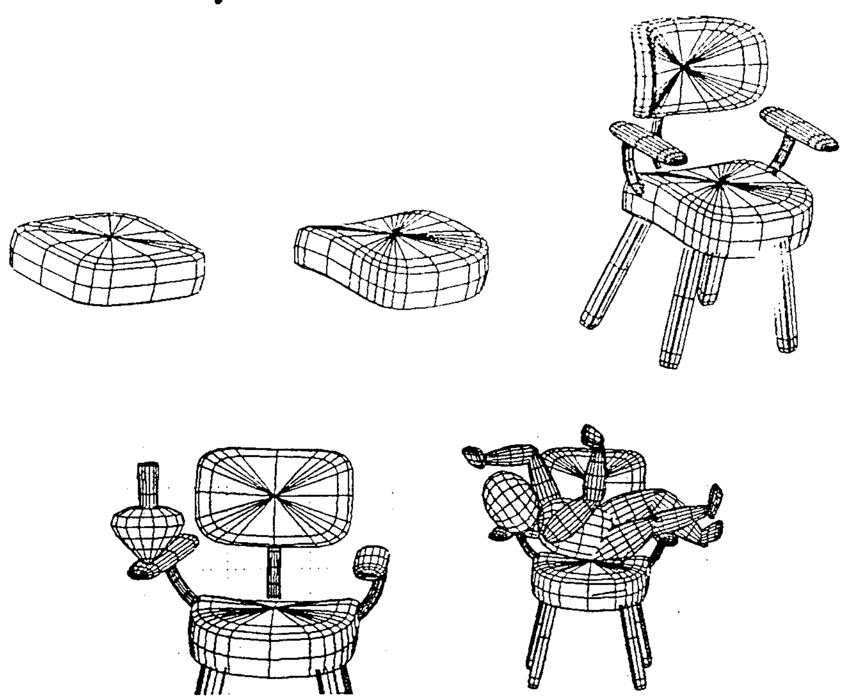

Figure 7 Design of A Chair Using Superquadric Functions to Represent Objects

REFERENCES

[1] Winner, R., Pennel, J., Bertrand, H., Slusarczuk, M., (1988), The Role of Concurrent Engineering in Weapons System Acquisition, Institute of Defense Analysis Report R–338

[2] Pentland, A., (1987) Towards an Ideal 3–D CAD System, *SPIE conference on Machine Vision and the Man–Machine Interface*, Jan. 11–16, San Diego, CA. SPIE Order No. 758–20

[3] Pentland, P.A. and Williams, J.R. (1989), Good Vibrations: Modal Dynamics for Graphics and Animation, ACM, SIGGRAPH 89, July 2–5, Boston, MA

[4] Hallquist, J.O., Goudrau, G.L. and Benson, D.J., (1985), Sliding Interfaces with Contact–Impact in Large–Scale Lagrangian Computations, Computer Methods in Applied Mechanics and Engineering, Vol. 51, pp 107–137

[5] Oden, J. T., and Kikuchi, N., (1982), Finite Element Methods for Constrained Problems in Elasticity, Int. Jour. for Num. Methods in Engineering, Vol. 18, pp 701–725.

[6] Bathe, K. and Chaudhary, A., (1985), A solution Method For Planar and Axisymmetric Contact Problems, Int. Jour. for Num. Methods in Engineering, Vol. 21, pp 65–88.

[7] Sutherland, I., (1963) Sketchpad: A Man Machine Graphical Communications System, Spring Joint Computer Conference, reprinted in Computer Graphics, H. Friedman, ed., IEEE Comp. Soc., 1980, pp 1–19

[8] Borning, A., (1970) Thinglab – A Constraint–Oriented Simulation Laboratory. SSL–79–3, Xerox PARC, Palo Alto, CA

[9] Terzopulos, D., Platt, J., Barr, A. and Fleischer, K., (1987) Elastically Deformable Models, Computer Graphics, Vol. 21, No. 4, pp 205–214

[10] Williams, J.R., Mustoe, G. G. W., and Hocking, G.,(1985) The Theoretical Basis of the Discrete Element Method, NUMETA 85, Swansea, U.K., Published in Numerical Methods in Engineering, Theory and Application, Rotterdam: Balkema Publishers

[11] Williams, J.R., (1988), Contact Analysis of Large Numbers of Interacting Bodies Using Discrete Modal Methods for Simulating Material Failure on the Microscopic Scale, Eng. Comput., Vol. 5, Sept.. pp 198–209

[12] Hallquist, J.O., (1978), A Numerical Treatment of Sliding Interfaces and Imapct, eds Park, and Gartling, Computational Techniques for Interface Problems, AMD, Vol. 30, ASME, New York

[13] Pentland, A. P., and Williams, J. R., (1989), Perception of Non–Rigid Motion: Inference of Shape, Material and Force, To appear in Int. Jour. Computer Artificial Intelligence.

[14] Barr, A.,(1984) Global and local deformations of solid primitives. Computer Graphics, 18, 3, pp. 21–30

[15] Pentland, A.,(1986) Perceptual Organization and the Representation of Natural Form, Artificial Intelligence Journal, Vol. 28, No. 2, pp. 1–38.

[16] Hanson, J.A., (1988), Hyperquadrics: Smoothly Deformable Shapes with Convex Polyhedral Bounds, Computer Vision, Graphics and Image Processing, 44, pp 191–210

[17] Gardiner, M., (1965) The superellipse: a curve that lies between the ellipse and the rectangle, Scientific America, Sept. 1965.

[18] Babuska, I., (1973), The Finite Element Method with Lagrangian Multipliers, Num. Mat., 20

[19] Oden, J.T., (1982), Penalty Finite Element Methods for Constrained Problems in Elasticity, Proc. Symp. finite Element Method, Science Press, Beijing:Gordon and Breach, New York, pp 150–197

[20] Hughes, T.J.R., Taylor, R.L. and Kanoknukulchai, W.A., (1979), A Finite Element Method for Large Displacement Contact and Impact Problems, in Formulations in Finite Element Analysis, eds Bathe, Oden and Wunderlich, US–German Symp. Finite Element Method